OCTOGON Mathematical Magazine
Vol. 33. No. 2A. October, 2025.

OCTOGON MATHEMATICAL MAGAZINE
Vol. 33, No.2, October 2025, pp 1366-1383
Print: ISSN 1222-5657, Online: ISSN 2248-1893
https://web.uni-miskolc.hu/∼matsefi/Octogon/

Designing for responsible physics and sciences with application to quantum Shannon entropy: An alternative algorithm for Ultra-Efficient Video Compression

Florentin Smarandache and Victor Christianto [32]

Dedicated to Themistocles M. Rassias on the occasion of his 75^{th} Birthday

ABSTRACT. It is known, that many if not most less-developed and under-developed countries still find problems in relation to digital infrastructures, and probably it is quite safe to say that many countries including those in Asia are having mobile telecom network still around 3.0-4.0 G [cf. 5-6].

Therefore, in order to improve their network capacity, we consider it would be a very interesting path to consider an algorithm solution to ultra-efficient compression, in order to allow big files such as media imaging, video etc can be transmitted at the existing mobile infrastructures, instead of developing large infrastructure including cell towers etc to go to 4.5 G or 5G etc. [1][2][3]
Such an alternative method that we discuss here, was inspired among other things by suggestions by R.N. Boyd, PhD., a senior physicist, and also for instance by a philosophy of Indian govt to "left offer (responsible) software solutions to infrastructure problems"right (that is at least what we read from several publications; although of course not all infrastructure problems can be solved entirely with software approach).
The pursuit of ultra-efficient video compression, pushing far beyond the limits of current standards like H.265 or AV1, requires a fundamental re-evaluation of how we define and transmit digital information [3].
This exploration ventures into the realm of quantum information theory [7] and advanced mathematical concepts, proposing a new paradigm that could theoretically achieve compression ratios exceeding 1000:1.
The essence of this argument rests on two novel and interconnected ideas: a modified form of quantum Shannon entropy and the application of the Argand complex plane [1][2].

[32]Received: 23.09.2025
2010 *Mathematics Subject Classification.* 33D80.
Key words and phrases. Entropy.

INTRODUCTION

The Limitations of Classical Information Theory

The pursuit of ultra-efficient video compression, pushing far beyond the limits of current standards like H.265 or AV1, requires a fundamental re-evaluation of how we define and transmit digital information.

This exploration ventures into the realm of quantum information theory and advanced mathematical concepts, proposing a new paradigm that could theoretically achieve compression ratios exceeding 1000:1 [cf. 1,2,3,7].

Traditional video compression is built upon the foundation of Claude Shannon's classical information theory. This framework defines a " bit" as the fundamental unit of information, representing a binary choice (0 or 1). Compression algorithms, whether lossless or lossy, work by removing redundancy and exploiting perceptual limitations within this binary framework [2][3]. They identify patterns, predict future frames, and discard data that is deemed less important to the human eye.

While highly effective, this approach has inherent limitations. The Shannon-Fano theorem, for instance, establishes a theoretical lower bound on the number of bits required to represent a symbol, a limit we are constantly approaching but can't surpass within the classical model. To achieve a 1000:1 compression ratio, we must abandon the classical bit as the indivisible unit of information [cf. 7].

Considering Software-Led Approach to Infrastructure

As we mentioned above in abstract section, "left a philosophy of Indian govt to "left offer (responsible) software solutions to infrastructure problems"right (that is at least what we read from several publications; although of course not all infrastructure problems can be solved entirely with software approach)."left

The Indian government is increasingly adopting a philosophy of using

responsible software solutions to address complex infrastructure challenges.

This approach acknowledges that while physical infrastructure is essential, digital platforms and data-driven systems can significantly enhance its efficiency, accessibility, and transparency. This isn't about replacing physical infrastructure with software, but rather about creating a complementary " digital layer" that makes traditional infrastructure more intelligent, coordinated, and effective, all while embedding principles of accountability and inclusion.

The essence of the idea is that many of the hurdles in traditional infrastructure projects—like a lack of coordination, bureaucratic delays, and uneven service delivery—can be mitigated or solved through smart, scalable, and secure digital systems.

Software solutions are used to manage a wide array of urban services, from traffic control and waste management to public transport. For instance, intelligent traffic systems use data analytics to dynamically adjust signal timings based on real-time traffic flow, effectively increasing the capacity of existing roads.

These software tools make urban living more efficient and responsive, while also providing a framework for citizen engagement and better municipal governance.

However, the " responsible" aspect of this philosophy is crucial. The Indian government's approach is not just about building tech for the sake of it, but about creating Digital Public Goods (DPGs) that are open, secure, and built for public benefit.

This includes developing frameworks for data protection and privacy, ensuring that these systems are transparent and accountable. This focus on responsibility is essential for building public trust and ensuring that these powerful digital tools serve the welfare of all citizens, particularly the most vulnerable, and do not lead to unintended consequences like surveillance or exclusion.

In conclusion, the Indian government's strategy of using responsible software solutions is a modern and pragmatic way to tackle age-old infrastructure

It acknowledges the limitations of a purely physical approach and seeks to leverage the scalability and efficiency of digital technology.

Learning from Schumacher and Sen: Designing for responsible physics and sciences

The Indian government's philosophy, to mention a few, of responsible software solutions, as a way to address infrastructure problems, can be re-interpreted through the lenses of E.F. Schumacher's concept of " appropriate technology" and Amartya Sen's " capabilities approach."

This framework suggests that technology should not be imposed from the top down, but should be human-centered, affordable, and empowering.

Schumacher advocated for " appropriate technology," which is defined by being small-scale, locally sustainable, and accessible to the masses. Schumacher believed that technology should not displace labor or require massive capital investment that is often unaffordable and dependent on foreign aid. This philosophy is directly relevant to the debate around 5G and 6G networks. These technologies require immense capital expenditure for new towers, fiber optics, and spectrum licenses, making them largely inaccessible to a majority of the population in less-developed countries. They represent a " bigger is better" approach that often bypasses the most critical needs of the poor. A Schumacherian perspective would question the wisdom of investing trillions in these networks when a large portion of the population lacks basic digital access.

Besides, Amartya Sen's capabilities approach provides a powerful ethical framework to evaluate this issue. Sen argues that the goal of development should not be merely an increase in GDP or technological output, but an expansion of people's " substantive freedoms" —the real opportunities they have to do and be what they value. From this perspective, a technology like 5G or 6G isn't inherently good; its value is determined by whether it actually expands people's capabilities. If a village cannot afford a 5G connection or a compatible device, the technology provides them with no freedom to learn, connect, or participate in the digital economy. The capabilities approach shifts the focus from the technology itself to the people

it is meant to serve, asking: " Does this technology actually enable people to lead lives they have reason to value?"

Combining the insights of Schumacher and Sen leads to a powerful framework for designing responsible technology solutions. It suggests that such solutions should be affordable and accessible (Schumacher), and they must demonstrably expand human capabilities (Sen).

In the context of connectivity, this means prioritizing technologies that are low-budget, easy to maintain, and provide a direct path to valuable functions like education, healthcare, and economic participation (in essence, to put development as if people mattered [8][9]). This is where a new approach to data compression comes into play. The sheer volume of data required for high-definition video and large files is the main driver behind the need for expensive, high-bandwidth 5G and 6G networks. If we can dramatically reduce the size of this data, we can make existing, more affordable networks like 4G and even 3G highly efficient, thereby extending their useful life and reach.

Introducing Quantum Information: Quantum Shannon Entropy and a New Path for Connectivity

This is where the application of modified quantum Shannon entropy becomes a fascinating and promising area of research. While conventional Shannon entropy defines the theoretical limit of data compression based on classical probability, quantum entropy operates on a different, more nuanced set of principles.

By leveraging the unique properties of quantum mechanics, a modified quantum compression algorithm could potentially achieve a compression ratio of 1000:1 for high-definition video and other data-intensive files. Such a breakthrough would be a game-changer for less- developed countries.

The logical step forward is to adopt the principles of quantum mechanics. In quantum information theory, the fundamental unit is the qubit (quantum bit). Unlike a classical bit, which exists in a definite state of 0 or 1, a qubit can exist in a superposition of both states simultaneously.

This property, along with entanglement, allows a single qubit to encode

exponentially more information than a classical bit [7].

For example, while a system of two classical bits can be in one of four possible states (00, 01, 10, 11), a system of two qubits can be in a superposition of all four states at once. This inherent informational density of the qubit is the first key to unlocking ultra-efficient compression.

The Path Altered: Is it viable to consider Making 4G Work Like 6G?

With such a dramatic compression ratio, a 1000:1 reduction in file size would mean that a video that currently requires a high-speed 5G connection could be streamed smoothly over a much slower and more widely available 4G network. This would effectively " alter the pathway" to next-generation connectivity.

Instead of requiring massive capital investments in 5G and 6G infrastructure that bypass rural and low-income populations, countries could focus on optimizing their existing networks. The focus would shift from building new, expensive physical infrastructure to developing innovative, people-centred software solutions that make the most of what is already there.

This technological approach is a perfect embodiment of the Schumacher and Sen philosophies. It is appropriate because it is not capital-intensive, it's scalable, and it can be implemented with minimal physical disruption.

It is responsible because it expands capabilities. It provides access to high-quality information and content for people who would otherwise be left behind by the high costs of 5G and 6G. This approach ensures that the benefits of the digital revolution are distributed more equitably, rather than concentrating them in urban centres and among the affluent. It is a model of development that is not only technologically sound but also socially just.

However, implementing such a radical solution is not without its challenges. The development of a functional and scalable quantum compression

algorithm is a monumental scientific and engineering task.

Furthermore, its widespread adoption would require collaboration between governments, tech companies, and civil society. There would be significant resistance from telecommunications companies and equipment manufacturers who have invested heavily in the 5G and 6G paradigm. The challenge is not just technological, but also political and economic. It requires a shift in mindset from a " big techies" approach to one that values efficiency, equity, and human well-being above all else. This is where the enduring philosophies of Schumacher and Sen must guide our path.

Discussion

Redefining Entropy for Quantum States

In classical information theory, Shannon entropy quantifies the uncertainty or randomness of a data source. A high-entropy source is difficult to compress. Our proposed model introduces a Modified Quantum Shannon Entropy (SQ), a theoretical construct that goes beyond simply calculating the entropy of a string of classical bits. This new entropy measure would calculate the inherent informational " footprint" of the video's quantum state.

The key modification is the consideration of perceptual redundancy in quantum terms. A video frame, rather than being a collection of pixels with RGB values, is conceptualized as a complex quantum state. This state includes not only the visual data but also its inherent statistical and temporal correlations. The Modified Quantum Shannon Entropy would measure the informational content of this quantum state, not the individual classical bits that describe it. For example, a vast, unchanging blue sky in a video frame has a very low classical entropy. In a quantum model, this could be represented as a single, low-energy quantum state. As the video changes, the quantum state evolves, and the algorithm would encode only the change in the state, not the entire new state itself.

The Argand Complex Plane Method

a. Visualizing Video Data in the Complex Plane

The second, and equally critical, component is the application of the Argand complex plane. Traditionally used to plot complex numbers, where each

number has a real and an imaginary part, this method offers a powerful new way to represent and manipulate video information.

In our model, each pixel, or even a group of pixels, is mapped to a single point on the Argand plane. The real axis could represent a primary visual attribute, such as luminance (brightness), while the imaginary axis could represent a secondary attribute, such as chrominance (color). For example, a bright red pixel might be mapped to a point far out on the positive real axis with a specific imaginary component.

b. Compressing Motion and Information Flow

The true power of this method lies in how it handles motion and temporal changes. Instead of storing a sequence of static frames, the algorithm would encode the flow or trajectory of points on the Argand plane over time. A single, smoothly moving object would not be a series of new pixel values but a continuous curve on the complex plane. This curve can be described by a simple mathematical function or a small number of parameters, achieving immense compression. For example, a car moving across the screen could be represented as a simple vector or a polynomial function on the Argand plane. This approach moves from a frame-by-frame representation to a motion-and-change-based representation. The Argand plane becomes a dynamic canvas where the video is not a series of snapshots, but a single, evolving trajectory of points.

A Unified Algorithm

The final step is to combine these two concepts into a unified compression algorithm. The process would look something like this:

1. Quantum State Representation: The raw video feed is first converted into a quantum state. This is not a physical process but a theoretical abstraction. The algorithm identifies and removes redundant classical information and represents the core informational " essence" of the video.

2. Mapping to the Argand Plane: This quantum state is then mapped onto the Argand complex plane. Each unique informational entity (e.g., a

visual object, a region of color) is given a position on the plane.

3. Entropy-Driven Encoding: The algorithm, guided by the Modified Quantum Shannon Entropy, only encodes the changes in the position of these points over time. The " cost" of encoding a change is determined by the new entropy of the system. Small changes, which correspond to low entropy, are represented by very few bits (or even " qubits"). Large, sudden changes in the video would result in a larger informational footprint on the Argand plane, requiring more bits to encode.

4. Reconstruction: On the decoder side, the algorithm would take the Argand plane data and reconstruct the visual representation, effectively reversing the process.

This hypothetical framework argues that by moving beyond the classical bit, embracing the information density of the quantum state, and using the Argand plane to represent and encode motion as a continuous flow rather than discrete changes, we can theoretically achieve compression ratios far beyond what is currently possible. It's an approach that doesn't just remove redundancy, but fundamentally redefines what constitutes the core information of a video.

Mathematica modelling (outline only)

Alternative 1. Mathematica code, based on Argand plane

(* ::Section::Closed::*)(*2. Argand-plane quantum-style compression (toy demo)*)

(*---*)
(*2.1 Helper:pack a byte list into complex numbers*)
(*---*)

bytesToComplex[byteList_List]:=Module[{pairs},
pairs=Partition[byteList,2,2,{1,1},{0,0}];(*pad with 0 if odd length*)Complex@@@pairs]

```
(*-------------------------------------------------------------------*)
(*2.2 Helper:unpack complex numbers back to bytes*)
(*-------------------------------------------------------------------*)

complexToBytes[complexList_List]:=Module[{realImag},
realImag=List@@#&/@complexList;
Flatten[realImag]]

(*-------------------------------------------------------------------*)
(*2.3 Load a small binary payload (here a short audio snippet)*)
(*-------------------------------------------------------------------*)

raw=Import[" ExampleData/sound.wav" ," Byte"];(*raw bytes of a wav
file*)Print[" Original size: " ,Length[raw]," bytes"];

(*-------------------------------------------------------------------*)
(*2.4 Map to complex plane*)
(*-------------------------------------------------------------------*)

z=bytesToComplex[raw];

(*-------------------------------------------------------------------*)
(*2.5 Apply a unitary transform–we use a normalized Hadamard matrix*)
(*H_n=(1/√n)*Walsh matrix (orthogonal,unitary over ℂ)*)
(*-------------------------------------------------------------------*)

n=Length[z];
(*Pad to nearest power of two for a perfect Hadamard matrix*)
pow2=2^Ceiling[Log2[n]];
pad=ConstantArray[0+0 I,pow2-n];
zPadded=Join[z,pad];
h=HadamardMatrix[Log2[pow2]]/Sqrt[pow2];(*orthogonal,
real→unitary*)zTrans=h.zPadded; (*forward unitary*)
```

```
(*------------------------------------------------------------------*)
(*2.6 Energy compaction– keep only the largest—z—values*)
(*------------------------------------------------------------------ *)

energy=Abs[zTrans]^2;
totalEnergy=Total[energy];
threshold=0.999;(*retain 99.9 % of total
energy*)sortedIdx=Ordering[energy,-All];(*descending
order*)cumEnergy=Accumulate[energy[[sortedIdx]]];
keepCount=FirstPosition[cumEnergy,_?(#¿=threshold*totalEnergy&)][[1]];
Print[" Keeping " ,keepCount," of " ,pow2," coefficients (≈ "
,NumberForm[keepCount/pow2*100, {4,1}],"%)"];
mask=ConstantArray[0+0 I,pow2];
Do[mask[[sortedIdx[[i]]]]=zTrans[[sortedIdx[[i]]]],{i,keepCount}];
zSparse=mask;(*most entries are exact zeros*)

(*------------------------------------------------------------------*)
(*2.7 Quantise polar components*)
(*------------------------------------------------------------------*)

(*Phase quantisation:8-bit uniform bins*)phases=Arg[zSparse];
phaseBits=8;
phaseLevels=2^phaseBits;
qPhase=Round[(phases+Pi)/(2 Pi)*(phaseLevels-1)]/(phaseLevels-1)*2
Pi-Pi;
(*Magnitude quantisation:keep as float (could be further reduced)*)
mag=Abs[zSparse];
(*optional:coarse-grain magnitudes,e.g.4-bit linear quantisation*)
magBits=4;
magLevels=2^magBits;
qMag=Round[mag*magLevels]/magLevels;
zQuant=qMag*Exp[I*qPhase];
```

```
(*-----------------------------------------------------------------*)
(*2.8 Serialize the sparse representation*)
(*-----------------------------------------------------------------*)

(*Store:(index,quantised magnitude,quantised phase) for non-zero coeffs*)
indices=Flatten@Position[zQuant,_?(#=!=0+0 I&)];
payload=Table[{indices[[k]],IntegerDigits[Round[qMag[[indices[[k]]]]*magLevels],
2,magBits],Integer Digits[Round[(qPhase[[indices[[k]]]]+Pi)/(2
Pi)*(phaseLevels-1)], 2,phaseBits]},{k,Length[indices]}];
(*Approximate compressed size (bits)*)
compressedBits=Length[indices]*Log2[pow2]+Length[indices]*magBits+
Length[indices]*phaseBits;
Print[" Compressed size ≈ " ,compressedBits," bits ("
,NumberForm[compressedBits/8,{5,2}]," bytes)"];

(*-----------------------------------------------------------------*)
(*2.9 Decompression–reconstruct the complex vector*)
(*-----------------------------------------------------------------*)

recon=ConstantArray[0+0 I,pow2];
Do[{idx,magBitsArr,phaseBitsArr}=payload[[k]];
magVal=FromDigits[magBitsArr,2]/magLevels;
phaseVal=FromDigits[phaseBitsArr,2]/(phaseLevels-1)*2 Pi-Pi;
recon[[idx]]=magVal*Exp[I*phaseVal];,{k,Length[payload]}];

(*Inverse unitary transform*)
zRec=ConjugateTranspose[h].recon;

(*Trim padding and convert back to bytes*)
bytesRec=Take[complexToBytes[zRec],Length[raw]];

Print[" Reconstruction error (L2 norm): " ,Norm[bytesRec-raw]/Norm[raw]];
```

The above algorithm of file compression, based on complex Argand plane, while it is still an outline, can be transformed into more real plan, for example using combined PHP 7 and Node.js framework (see Appendix I for

an outline of codes).

Alternative 2. Mathematica code (outline only).

Below is a runnable Mathematica notebook that shows a framework for a video-compression pipeline.

It incorporates:
1. Standard pre-processing (frame extraction, color-space conversion, block-wise DCT).

2. Quantization, to tune to aggressive levels.

3. A "left quantum-inspired"right entropy encoder that illustrates how one might embed a modified Shannon-entropy calculation on the complex-plane representation of DCT coefficients (the Argand diagram idea).

4. Reconstruction code, to verify visual quality versus file size.

Outline, Mathematica code

(* ::Package::*)(*Load necessary packages–these are built - in in recent versions of Mathematica*)Needs[" ImageProcessing‘"];

Needs[" Video‘"];(*Video import/export utilities*)Needs[" SignalProcessing‘"];(*DCT/IDCT functions*) (*Set a short alias for later use*)

ClearAll[quantTable];

(*Apply 2- D DCT to an 8×8 block (real numbers)*)blockDCT[blk_]:=DiscreteCosineTransform[blk,2];

(*Inverse DCT–returns the reconstructed block*) blockIDCT[coeff_]:=InverseDiscreteCosineTransform[coeff,2];

(*Standard JPEG- style luminance quantization matrix*)quantTable={{16,11,10,16,24,40,51,61}, {12,12,14,19,26,58,60,55},{14,13,16,24,40,57,69,56},{14,17,22,29,51,87,80,62},

```
{18,22,37,56,68,109,103,77},{24,35,55,64,81,104,113,92},
{49,64,78,87,103,121,120,101}, {72,92,95,98,112,100,103,99}];

(*Quantize an 8×8 DCT coefficient block*)
quantizeBlock[coeff_,scale_:1]:=Round[coeff/(quantTable*scale)];

(*De- quantize back to approximate DCT coefficients*)
dequantizeBlock[qblk_,scale_:1]:=qblk*(quantTable*scale);

(*Map a real coefficient to a complex number with a tiny
phase*)toComplex[coeff_]:=coeff+I*0.001*Sign[coeff];

(*Compute a modified entropy on a list of complex magnitudes*)

modifiedEntropy[complexList_,h
_:0.01]:=Module[{mag=Abs[complexList],p,H},(*Normalise to a probability
distribution*)p=mag/Total[mag];

(*Classical Shannon term plus a quantum - style correction*)H=-Total[p
Log[Max[p,10^-12]]] +h *Total[Log[mag+10^-12]];
H];

(*Encode a block by storing its entropy value (as a toy example)*)
encodeBlockEntropy[qblk_,h
_:0.01]:=Module[{cplx=toComplex/@Flatten[qblk]},
```

Concluding note

Important disclaimer – Achieving a 1 : 1000 compression ratio on typical video material is far beyond the state of the art. The aforementioned algorithm and codes a research-prototype scaffold. For further advancement, it is recommended to experiment with quantization tables, entropy-coding parameters, and possibly custom hardware to approach extreme ratios. The outline of Mathematica codes is for illustrative purpose only, it does not guarantee any particular compression factor. Further research in the

direction discussion herein is recommended [1][2].

Acknowledgment. These authors would extend gratitude to R. Neil Boyd, PhD for discussions, especially related to alternative to 5G, and ultra efficient algorithm for compression.
Version 1.0: 24 sept. 2025

REFERENCES

[1] S. Hill, Fady Alajaji, and Tamas Linder (2025). "left Communication Complexity of Exact Sampling under Renyi Information,"right arXiv. https://arxiv.org/abs/2506.12219v1
[2] S. Reddy, Anca D. Dragan, Sergey Levine (2021). Pragmatic Image Compression for Human-in-the- Loop Decision-Making. 35th Conference on Neural Information Processing Systems (NeurIPS 2021). url: https://proceedings.neurips.
cc/paper/2021/file/df0aab058ce179e4f7ab135ed4e641a9- Paper.pdf
[3] K. Sayood (2006). Introduction to Data Compression, 3rd ed. San Fransisco: Morgan Kaufmann Publishers.
[4] Edward A. Bender (1978). An introduction to mathematical modelling. New York: John Wiley & Sons, Inc.
[5] Zacc Ritter, Andrew Dugan, Chris Felton, Byron R. Johnson, R. Noah Padgett, Tyler J. VanderWeele ( ). "left WHAT CONTRIBUTES TO A LIFE WELL-LIVED?"right The Global Flourishing Study. Gallup, Inc.
[6] S. Naranjo Bautista, Sandra, E. Mosqueira, Edgardo, M. Lafuente (eds.) (2025). Better Governments for Better Lives: Strengthening State Capacities. Washington: Inter-American Development Bank.
[7] Wojciech H. Zurek (ed.) (1990). Complexity, Entropy & the Physics of Information. Santa Fe Institute Studies, in the Sciences of Complexity.
[8] Robert Leonard. E. F. Schumacher and Intermediate Technology. DOCUMENT DE TRAVAIL / WORKING PAPER, No. 2018-22
[9] N. Ornelas Martins. Sustainability and development through the humanistic lens of Schumacher and Sen. Ecological Economics (2022) 107532. DOI: https://doi.org/10.1016/j.ecolecon.2022.107532
[10] M. Rozahi Istambul & T. Abdulghani. Designing Inclusive Digital Platforms for Marginalized Communities: A Design Science Research Approach. Chapter in V. Mardiansyah and B. A. Prasetyo (eds.), Proceedings of the Widyatama International Conference on Engineering 2024 (WICOENG 2024), Advances in Engineering Research 252, https://doi.org/10.2991/978-94-6463-618-5_12

[11] C.P. Sianipar, G. Yudoko, K. Dowaki, and A. Adhiutama. Design Methodology for Appropriate Technology: Engineering as if People Mattered. Sustainability 2013, 5, 3382-3425; doi:10.3390/su5083382
[12] E.F. Schumacher. Keluar dari kemelut. Jakarta: LP3ES ( )
[13] Octogon Mathematical Magazine (1993-2025)

Appendix: using PHP and node.js to exemplify the above algorithm (outline only)

Alternative Approach: A Microservices-Based Solution

A more practical approach would be to build a system where each language handles what it's best at. Here is a conceptual breakdown of a possible solution.

1. The Node.js Backend (Computational Engine)

Node.js is well-suited for this task due to its asynchronous nature, which is good for handling mathematical calculations without blocking the server.

• Complex Number Class: create a Complex class to represent complex numbers, with methods for addition, subtraction, multiplication, division, and finding the argument.

• Endpoint for Calculation: Create a REST API endpoint (e.g., /calculate) that accepts the necessary parameters from the user (e.g., real and imaginary parts of numbers).

• Calculation Logic: Inside this endpoint, perform the complex number operations and calculations. The Mathematica algorithm would be translated into JavaScript code here.

• Visualization Data: The final output would be an array of complex

JavaScript
// A conceptual outline for the Node.js backend. const express = require('express'); const app = express(); const fs = require('fs').promises; //

Use async file I/O const math = require('mathjs'); // Use a library for complex numbers and matrices // Define a helper function to translate the Mathematica code. // For example, this is a simplified representation of the 'bytesToComplex' function. function bytesToComplex(byteList) { const complexArray = []; for (let i = 0; i ¡ byteList.length; i += 2) { // Math.js can handle complex numbers complexArray.push(math.complex(byteList[i], byteList[i + 1] —— 0)); } return complexArray; } // ... other helper functions like 'complexToBytes' ... app.post('/compress', async (req, res) =¿ { try { const raw = await fs.readFile('ExampleData/sound.wav'); const rawBytes = Array.from(raw); console.log('Original size: $ {rawBytes.length} bytes'); // 1. Map to complex plane const z = bytesToComplex(rawBytes); // 2. Apply a unitary transform (Hadamard matrix) const n = z.length; const pow2 = Math.pow(2, Math.ceil(Math.log2(n))); const zPadded = math.matrix([...z, ...Array(pow2 - n).fill(math.complex(0, 0))]); // You'd need to implement or find a Hadamard matrix function const hadamard = require('hadamard-matrix'); const h = math.multiply(hadamard(Math.log2(pow2)), 1 / Math.sqrt(pow2)); const zTrans = math.multiply(h, zPadded); // 3. Energy compaction and quantization // ... Implement the logic from the Mathematica code using a library like math.js // 4. Serialize the data const payload = []; const complexArray = zTrans.toArray(); // Convert math.js matrix to array for (let i = 0; i ¡ complexArray.length; i++) { if (complexArray[i].re !== 0 —— complexArray[i].im !== 0) { const z = complexArray[i]; const mag = math.abs(z); const phase = math.arg(z); // Store index, quantized magnitude, and quantized phase payload.push({ index: i, mag, phase }); } } // 5. Send back the compressed payload as JSON res.json({ payload }); } catch (error) { res.status(500).send(error.message); } }); app.listen(3000, () =¿ console.log('Node.js server listening on port 3000'));

The PHP7 Frontend: The Web Interface

PHP's role would be to provide the user with a web interface and act as a client that communicates with the Node.js backend. It would handle user requests, send them to the Node.js API, and display the results.

PHP7

// A conceptual outline for the PHP7 frontend. ¡?php // Function to send a request to the Node.js backend function compressFile($filePath)

{ $url = 'http://localhost:3000/compress'; $ch = curl_init($url); curl_setopt($ch, CURLOPT_RETURNTRANSFER, true); curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST'); curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode(['filePath' =¿ $filePath])); curl_setopt($ch, CURLOPT_HTTPHEADER, ['Content-Type: application/json']); $response = curl_exec($ch); $httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE); curl_close($ch); if ($httpCode !== 200) { throw new Exception(" Error from Node.js server: " . $response); } return json_decode($response, true); } // Check if a file has been uploaded if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_FILES['audioFile'])) { $filePath = $_FILES['audioFile']['tmp_name']; try { $compressedData = compressFile($filePath); echo " ¡h1¿Compression Result¡/h1¿" ; echo " ¡pre¿" . json_encode($compressedData, JSON_PRETTY_PRINT) . " ¡/pre¿" ; } catch (Exception $e) { echo " ¡h1¿Error¡/h1¿" ; echo " ¡p¿" . $e- ¿getMessage() . " ¡/p¿" ; } } ?¿ ¡!DOCTYPE html¿ ¡html¿ ¡head¿ ¡title¿Audio Compression¡/title¿ ¡/head¿ ¡body¿ ¡h1¿Upload Audio for Compression¡/h1¿ ¡form action="" method=" post" enctype=" multipart/form-data" ¿ ¡input type=" file" name=" audioFile" accept=" .wav" ¿ ¡button type=" submit" ¿Compress¡/button¿ ¡/form¿ ¡/body¿ ¡/html¿

Both, while having different features, form a microservices architecture. The Node.js application is a specialized service for the heavy-duty computation, and the PHP application is the user-facing web server that orchestrates the communication.

This separation of concerns allows each language to do what it's best at, leading to a more robust and maintainable system.

end of appendix

Dept. Mathematics and Sciences,
University of New Mexico, Gallup, NM, USA

Dept. Forestry, Malang Institute of Agriculture,
East Java, Indonesia